

Programming for Robotics



Newbury Park High School 2016

Intro to the easyC IDE

PFR.01.01

Created by LCpl Rice
20160701

- Navigating the IDE
- Recommended Settings
- Download Firmware to Robot
- Download Firmware to Joystick
- Test Cortex
- Upload Sample Code
- Communicate over wire
- Communicate over VEXnet
- Upload code over Joystick

Learning Objectives

Terminal = What you walk away with
Enabling = What you will be evaluated on

Terminal Learning Objectives

- Able to use multiple techniques to upload easyC programs and use VEXnet

Enabling Learning Objectives

- Able to recognize and use USB Type A in accordance with “IEC 62680”, *USB Implementers Forum*, 2013
- Able to recognize and use RJ12 and RJ45 cables in accordance with “name”, *creator*, year.
- Able to pair VEXnet devices in accordance with “VEXnet User Guide”, *VEX*, 2015
- Able to upload easyC code to VEX Cortex in accordance with “Programming With easyC”, *BEST Robotic, Inc.*, 2014

Terms & Abbreviations

- Terminal (robotics) - Text based output from the robot program
- Terminal (computers) - CLI where the user can type commands
- Debugging - Troubleshooting, or the process of finding and solving errors
- Cortex - The microcontroller of the VEX system, aka the brain.
- EDR - this version of VEX, the previous was PIC
- USB - Universal Serial Bus
- Ethernet Cable - A cable used (typically) for internet connections
- RJ12 - Registered Jack 12, similar to the RJ45 which is used for Ethernet

USB Types

Cell Phone chargers commonly use USB 2.0 Micro Type B

Printers commonly use USB 2.0 Type B

USB Extension Cables use USB Type A MALE to USB Type A FEMALE

VEX uses USB Type A MALE to USB Type A MALE

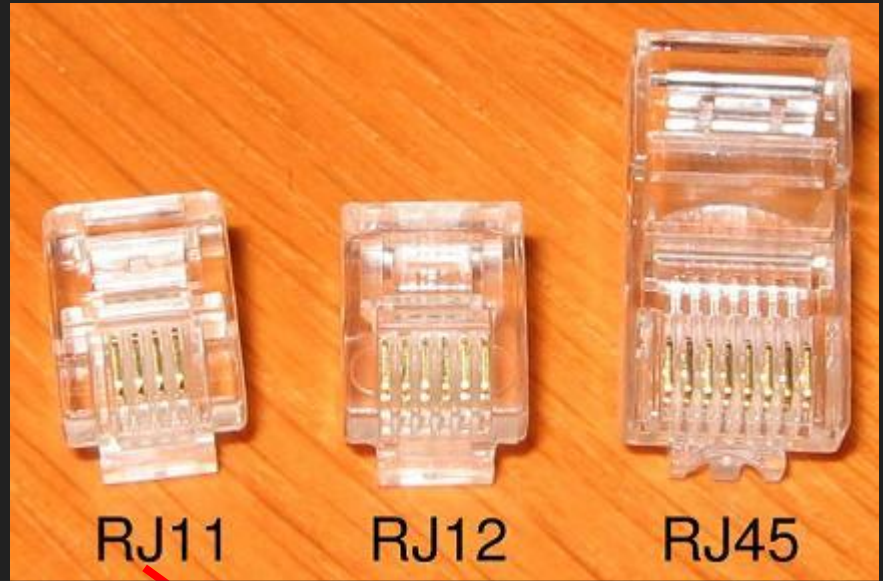
MALE	FEMALE	MALE	FEMALE
			
USB 2.0 Type A Plug	USB 2.0 Type A Jack	USB 3.0 Type A Plug	USB 3.0 Type A Jack
			
USB 2.0 Type B Plug	USB 2.0 Type B Jack	USB 3.0 Type B Plug	USB 3.0 Type B Jack
			
USB 2.0 Mini Type B Plug (4 Position)	USB 2.0 Type B Jack (4 Position)	USB 2.0 Micro Type B Plug	USB 2.0 Micro Type B Jack
			
USB 2.0 Mini Type B Plug (5 Position)	USB 2.0 Type B Jack (5 Position)	USB 3.0 Micro Type B Plug	USB 3.0 Micro Type B Jack

Registered Jacks

Ethernet (internet) cables use RJ45, and has 8 pins/wires in the cable.

The USB to Serial adapter uses RJ12 - a smaller connector with less pins than the RJ45

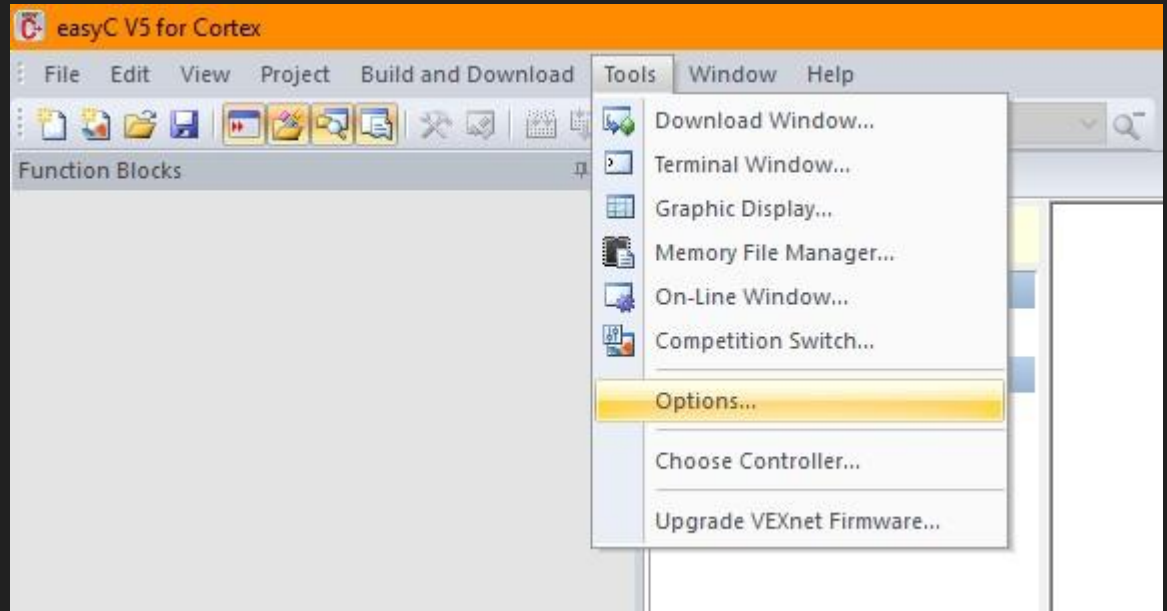
The RJ11 and the RJ12 are both used for telephone lines, though the RJ11 standard only uses 4 wires, and the RJ12 uses 6.



Navigating the IDE

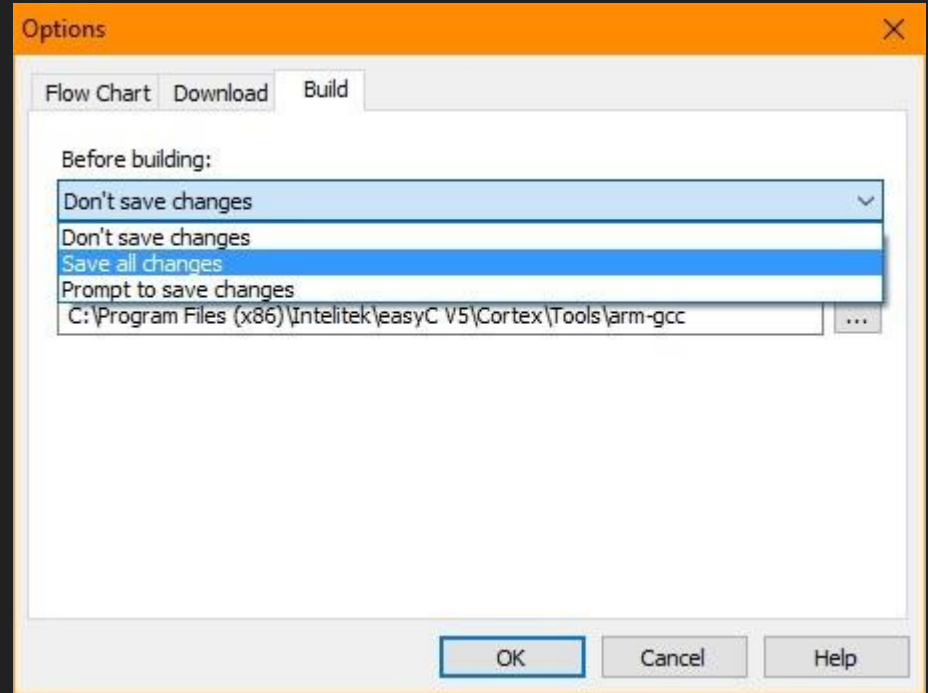
Recommended Settings

Click Tools -> Options



Recommended Settings

Select Build, then change the drop down box to “Save all changes” and click OK.



Download Firmware to Robot

You need to update firmware when switching between easyC and RobotC!

Download Firmware to Robot

For these steps, we need the VEX Cortex and USB A-A Cable as pictured:



Download Firmware to Robot

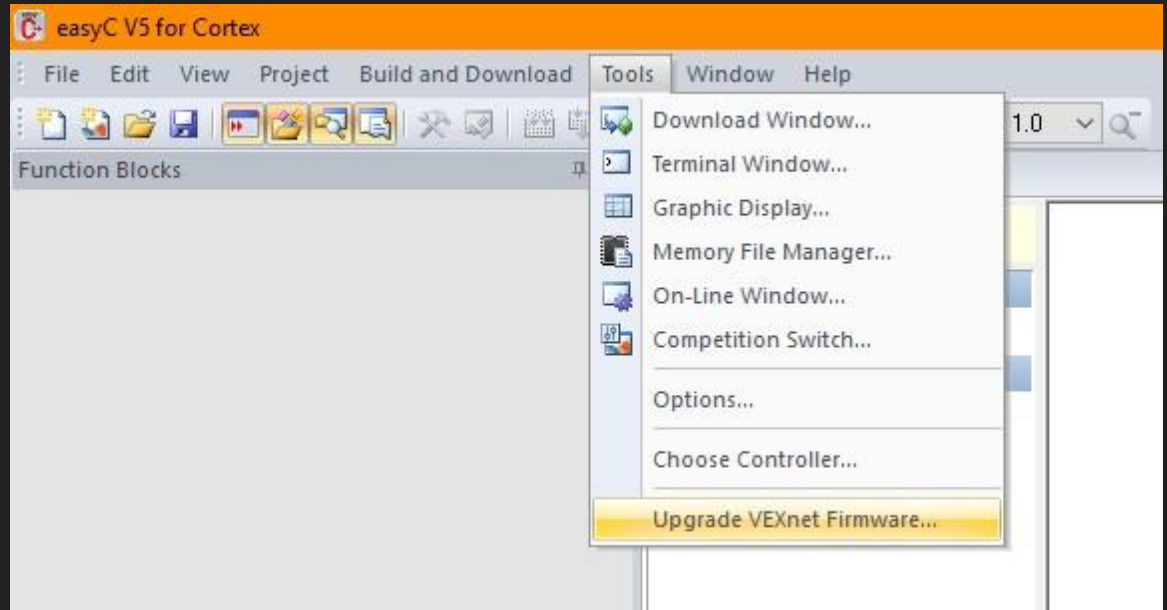
Connect the USB cable to the VEX Cortex (as pictured) and to a USB port on the computer with easyC running. If done correctly, “ROBOT” should be slowly blinking RED, and “VEXnet” should be solid GREEN.

NOTE: Cortex is turned OFF.



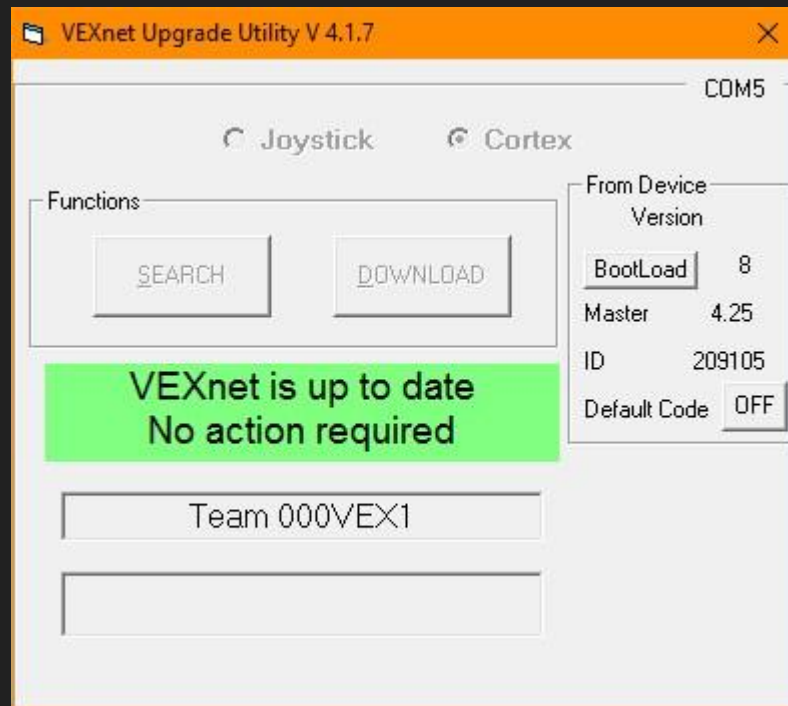
Download Firmware to Robot

In easyC, select Tools ->
Upgrade VEXnet
Firmware...



Download Firmware to Robot

If the window looks like this, then you are good! If not, follow the appropriate steps.



Download Firmware to Joystick

You need to update firmware when switching between easyC and RobotC!

Download Firmware to Joystick

For these steps, we need the VEX Joystick and USB A-A Cable as pictured:



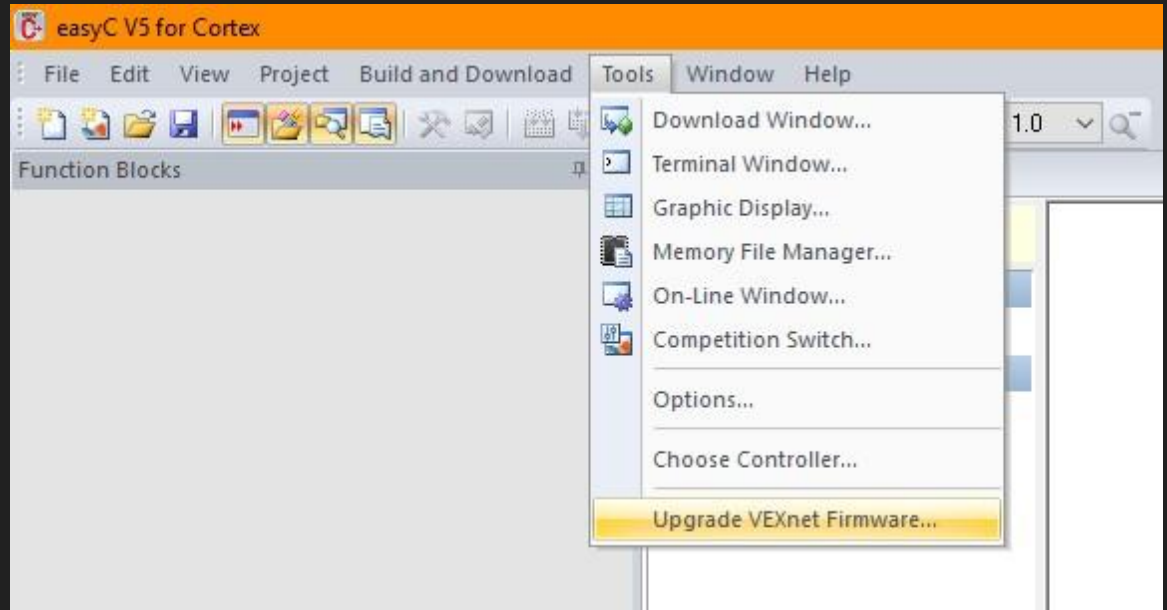
Download Firmware to Joystick

Connect the USB cable to the Joystick (as pictured) and to a USB port on the computer with easyC running. If done correctly, “JOYSTICK” should be solid RED, and “VEXnet” should be solid GREEN. **NOTE: Joystick is turned OFF.**



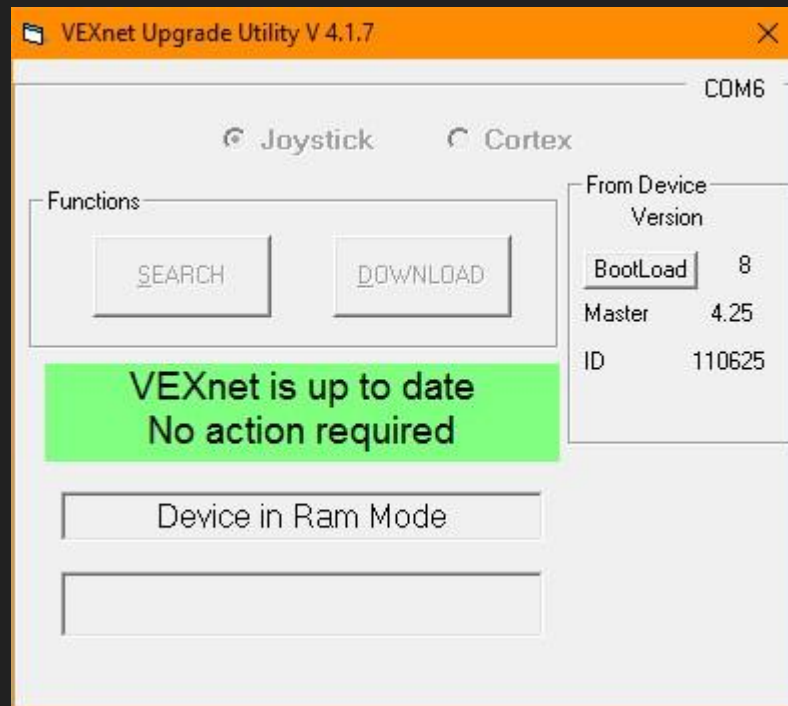
Download Firmware to Joystick

In easyC, select Tools ->
Upgrade VEXnet
Firmware...



Download Firmware to Joystick

If the window looks like this, then you are good! If not, follow the appropriate steps.



Test Cortex

Testing the cortex is useful to ensure that everything is working when you are trying to debug an issue. The following steps will show you how to make sure the cortex and have code uploaded to it, and run the code, and display messages on the terminal.

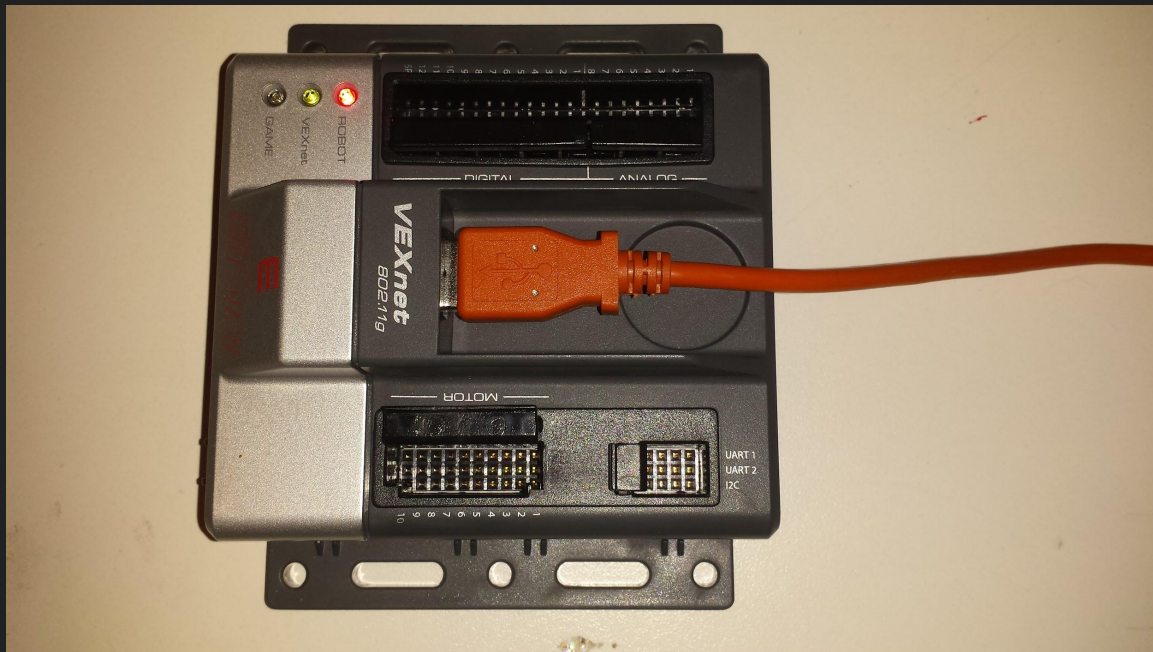
Test Cortex

For these steps, we need the VEX Cortex and USB A-A Cable as pictured:



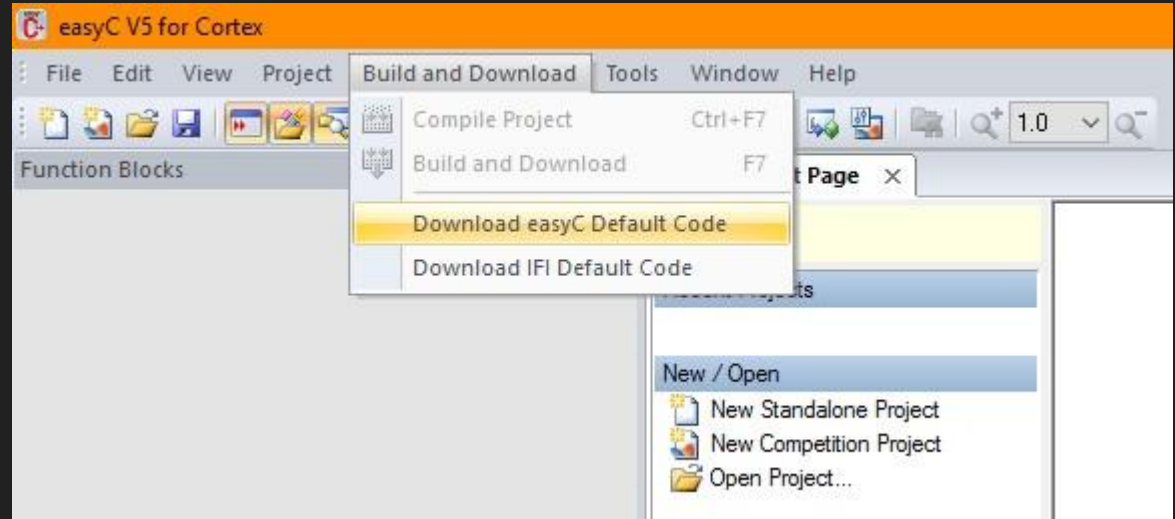
Test Cortex

Connect the USB cable to the VEX Cortex (as pictured) and to a USB port on the computer with easyC running. If done correctly, “ROBOT” should be slowly blinking RED, and “VEXnet” should be solid GREEN.



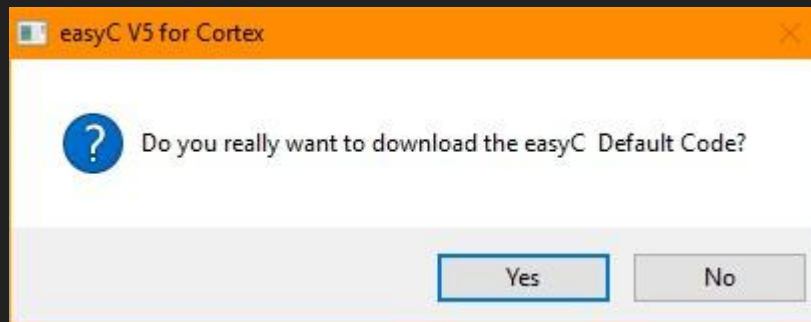
Test Cortex

In easyC, select Build and Download -> “Download easyC Default Code”



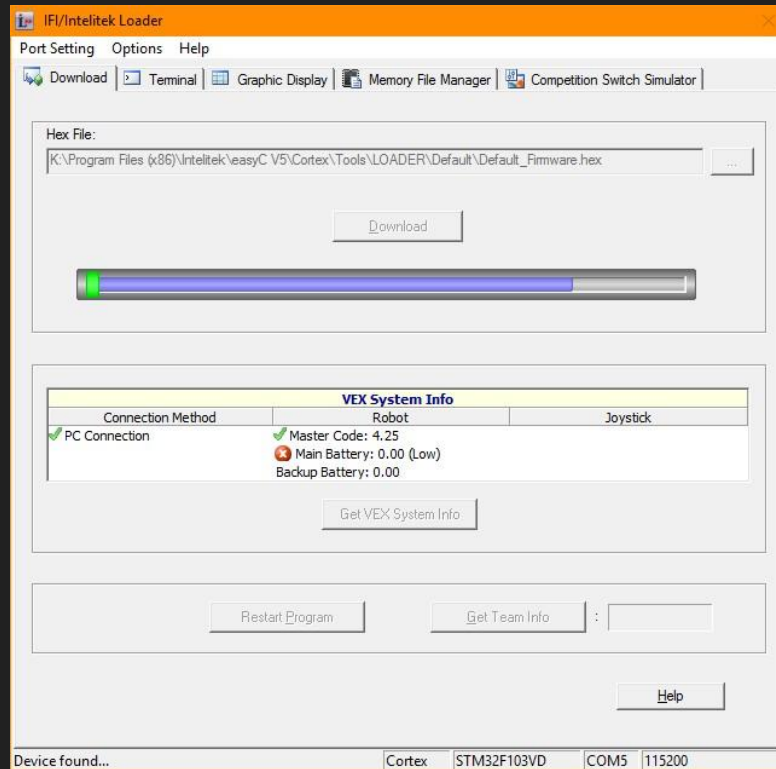
Test Cortex

Click Yes



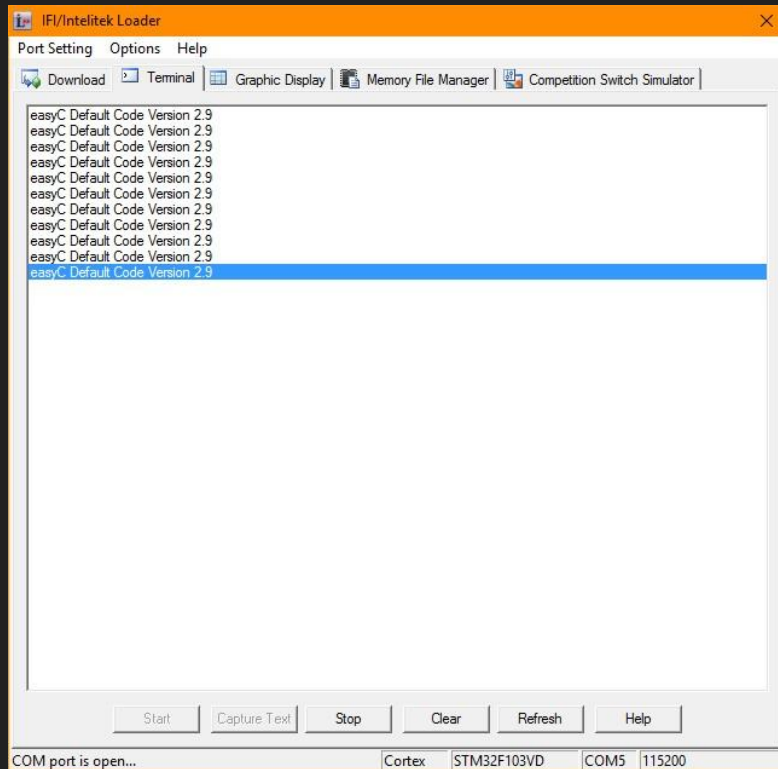
Test Cortex

This dialogue box should appear, wait for it to finish downloading.



Test Cortex

Once it has downloaded correctly, the Terminal should pop up and repeatedly show the text pictured. If you get that, then your cortex is working.



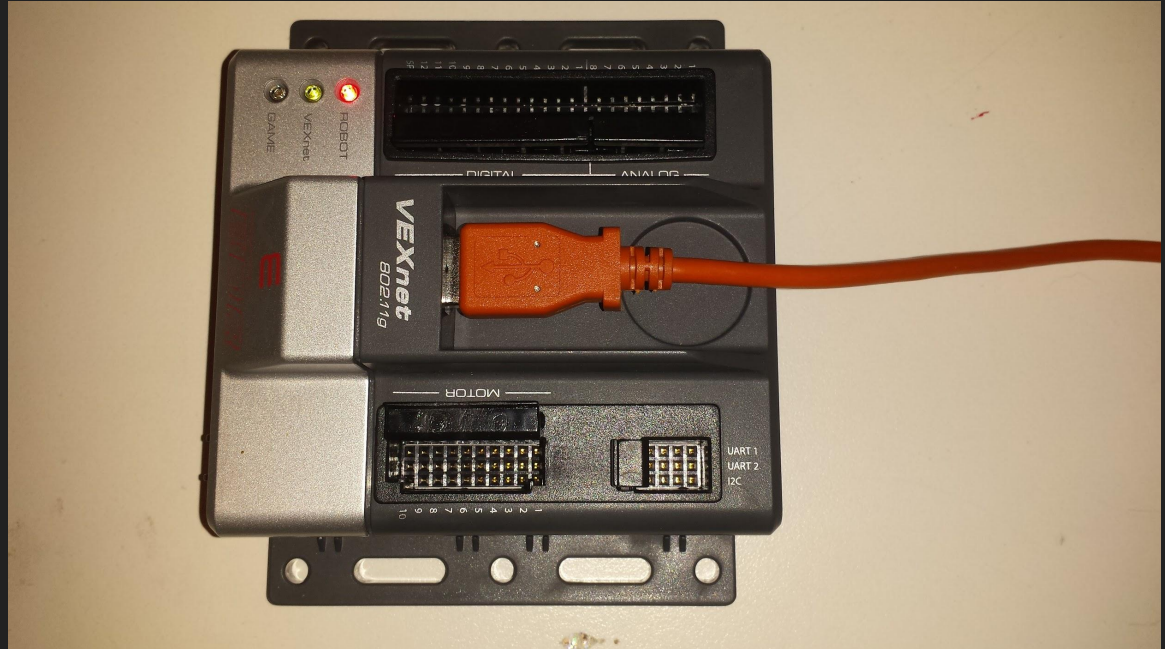
Upload Sample Code

For these steps, we need the VEX Cortex and USB A-A Cable as pictured:



Upload Sample Code

Connect the USB cable to the VEX Cortex (as pictured) and to a USB port on the computer with easyC running. If done correctly, “ROBOT” should be slowly blinking RED, and “VEXnet” should be solid GREEN.



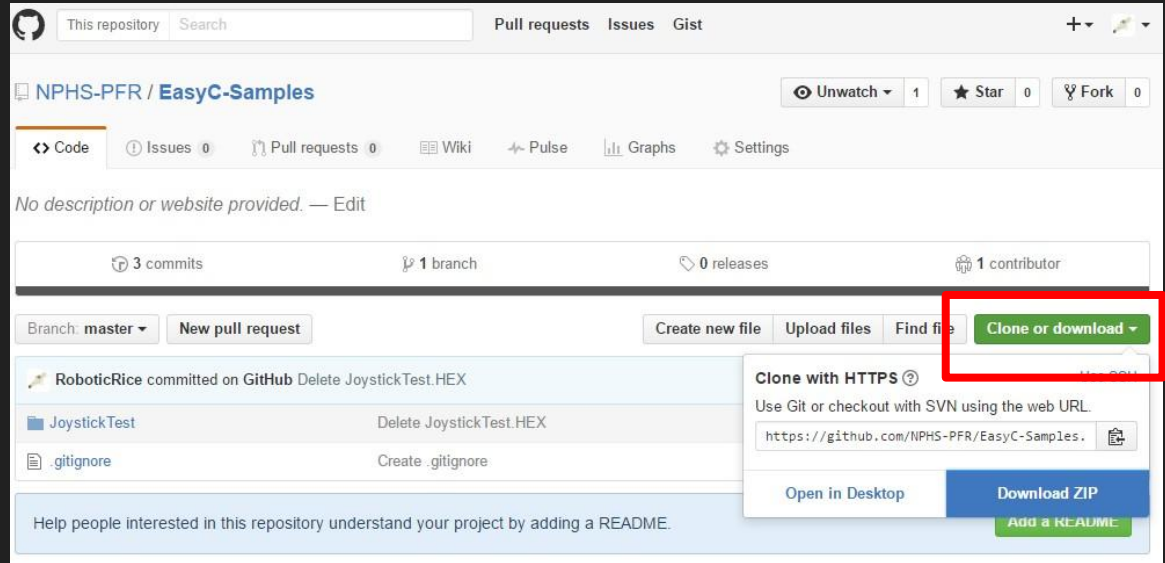
Upload Sample Code

Open an internet browser
and go to

<https://github.com/NPHS-PFR/EasyC-Samples>, click

“Clone or download” ->

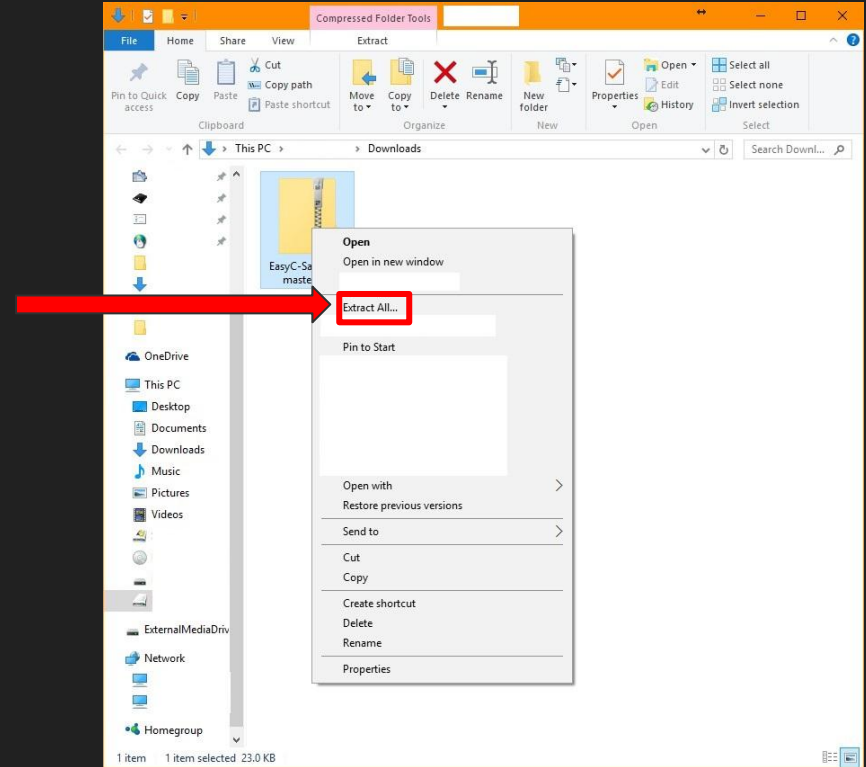
“Download ZIP”



Upload Sample Code

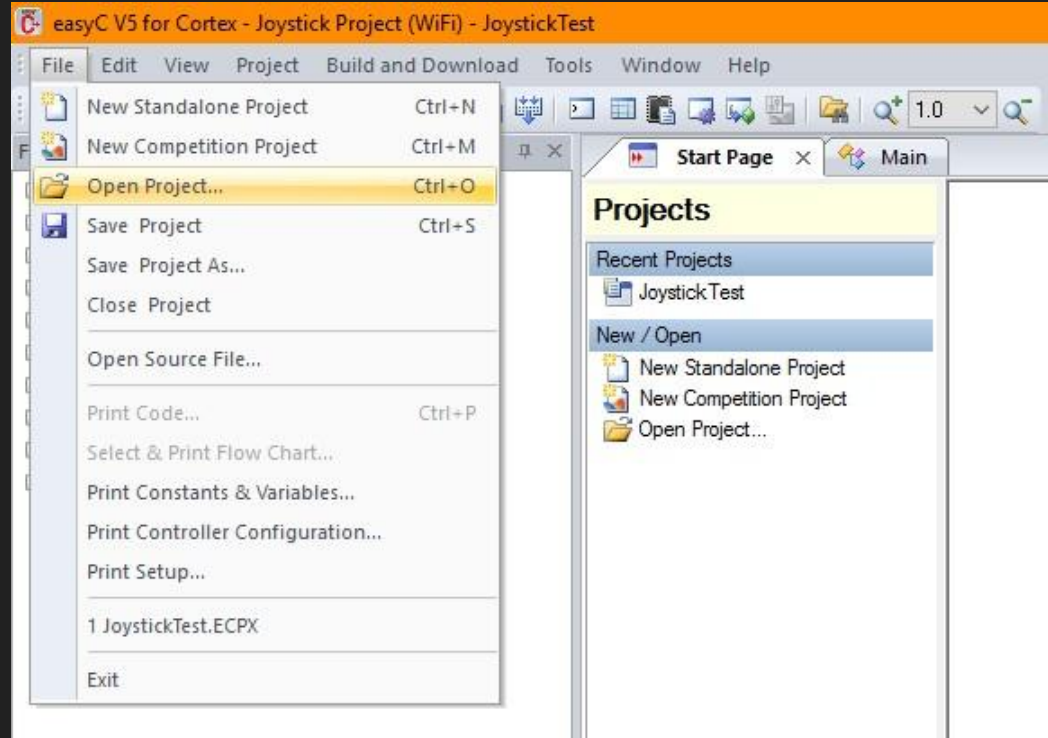
Right Click on the .zip file
and select “Extract All...”

Follow the prompts until the
files have been extracted.



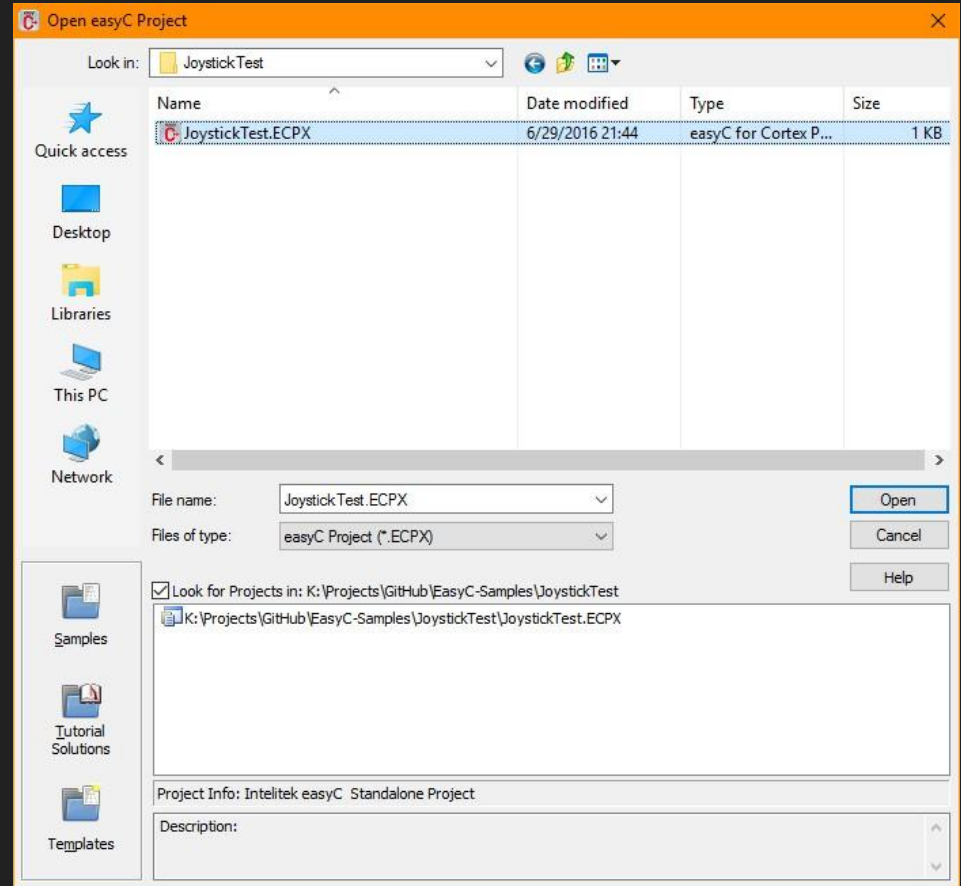
Upload Sample Code

Open up easyC V5 and go to File -> Open Project...



Upload Sample Code

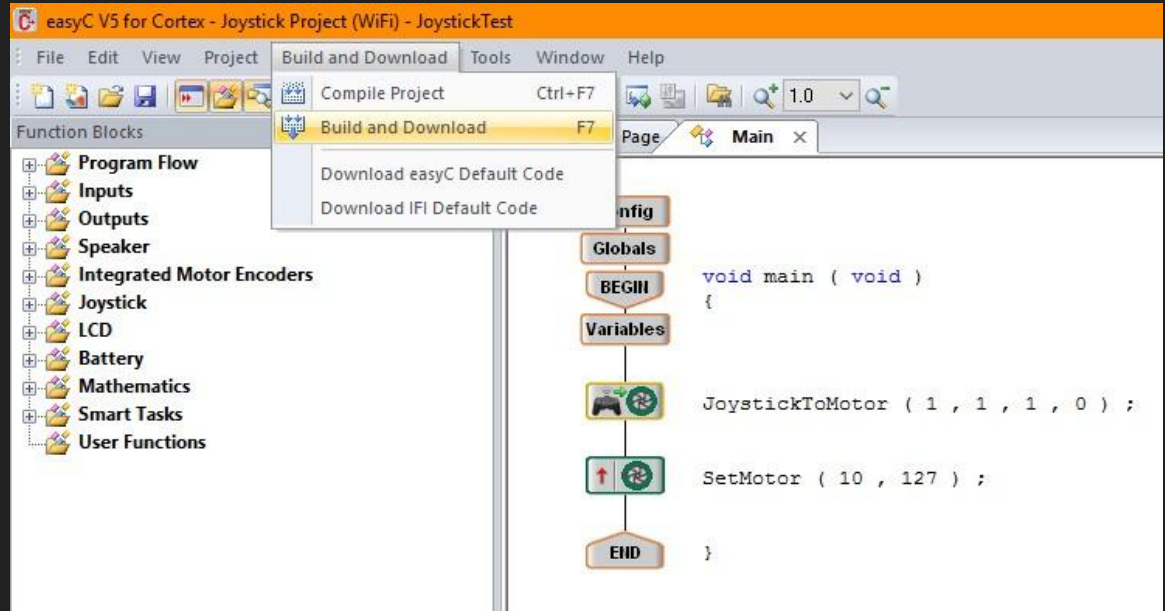
Navigate to the JoystickTest folder, and open up “JoystickTest.ECPX”



Upload Sample Code

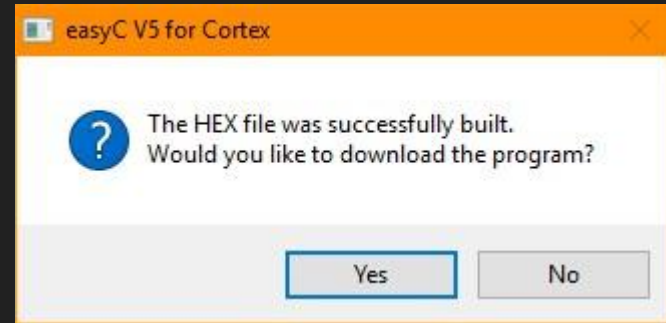
With the program loaded in easyC for VEX and the Cortex still connected to the computer, go to “Build and Download” -> “Build and Download”

(or you can hit F7)



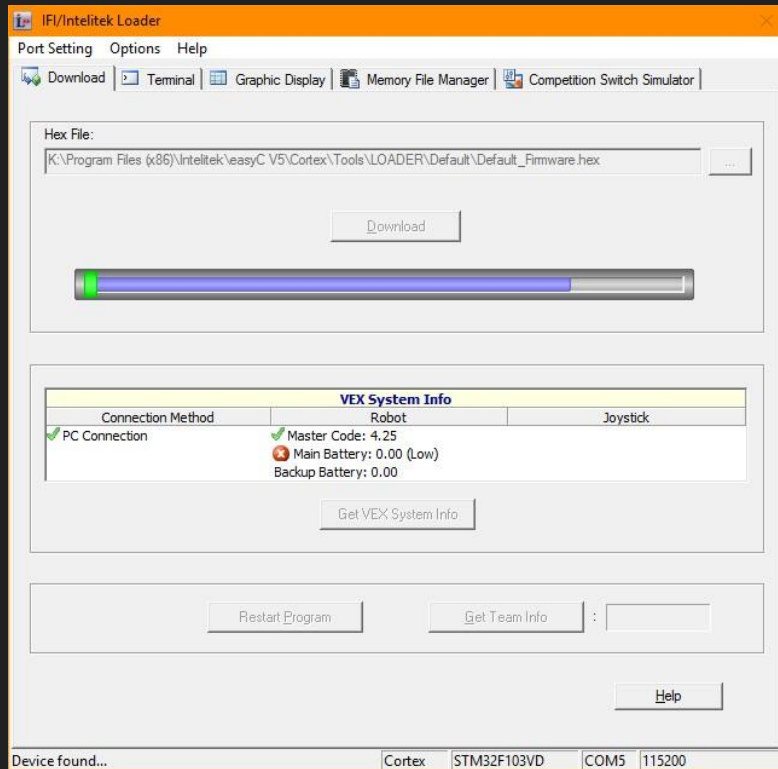
Upload Sample Code

If this box appears, click
YES



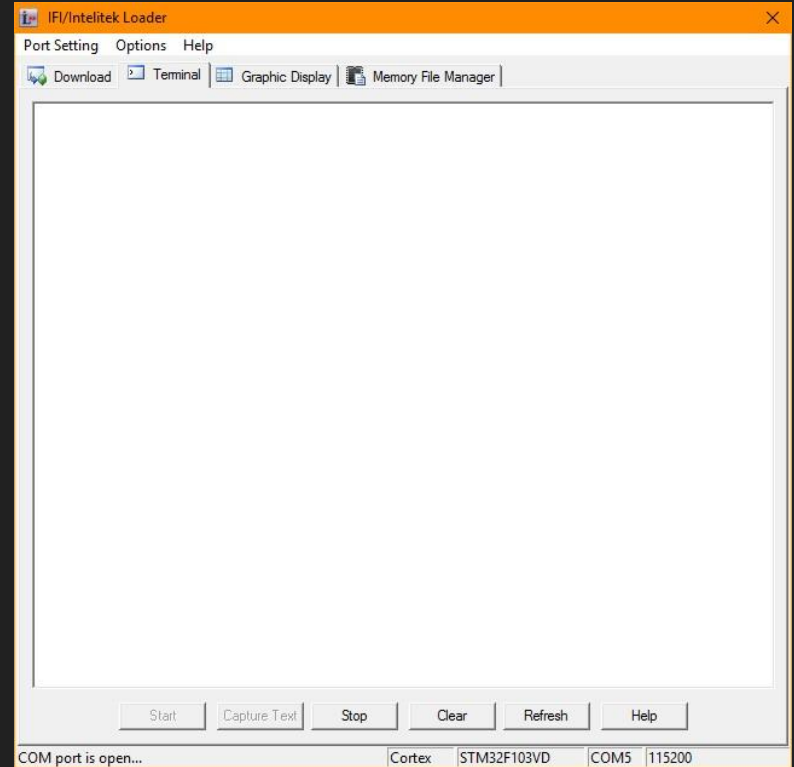
Upload Sample Code

This dialogue box should appear, wait for it to finish downloading.



Upload Sample Code

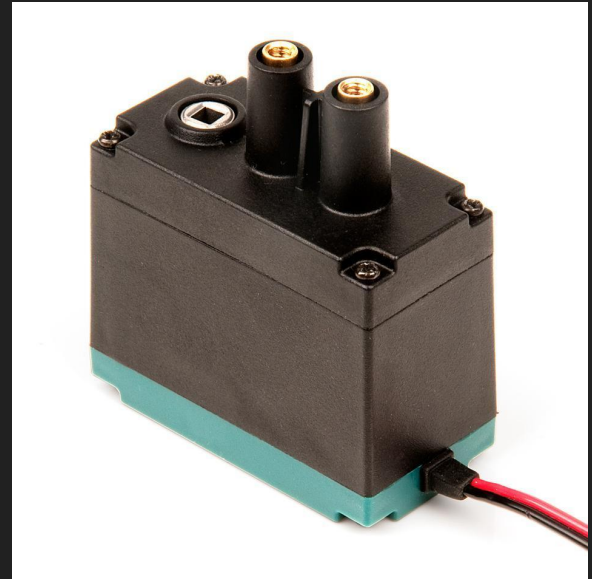
Once the terminal box opens, the program has been downloaded!



Communicate over wire

In order to operate the robot with a joystick, the joystick must either be connected to the Cortex via USB cable or Wifi - this will explain how to drive the robot with the JoystickTest program already installed. Please note that the specific program JoystickTest requires the use of one Two-Wire motor as pictured, but motors are not required to communicate between the Cortex and Joystick.

NOTE: Joystick does not require batteries for wired communications!



Communicate over wire

For these steps we need the VEX Cortex, Joystick, 7.2V Battery & USB A-A Cable



Communicate over wire

With the USB cable still plugged into the Cortex and the correct program already downloaded, plug in the 7.2V battery into the Cortex as pictured.



Communicate over wire

Unplug the USB cable from the computer (if you haven't already) and plug it into the Joystick.

NOTE: Joystick and Cortex should now be plugged into each other.



Communicate over wire

If using JoystickTest, plug the 2-Wire motor into port 1 (for joystick control) or port 10 (for continuous power).



Communicate over wire

Turn on the Cortex. It may take a couple seconds for the device to boot, but if it is working the VEXnet light on both the joystick and Cortex should be solid green.

Make sure the light is green for a solid 5 seconds to pair.

NOTE: Joystick does not need to be turned on!

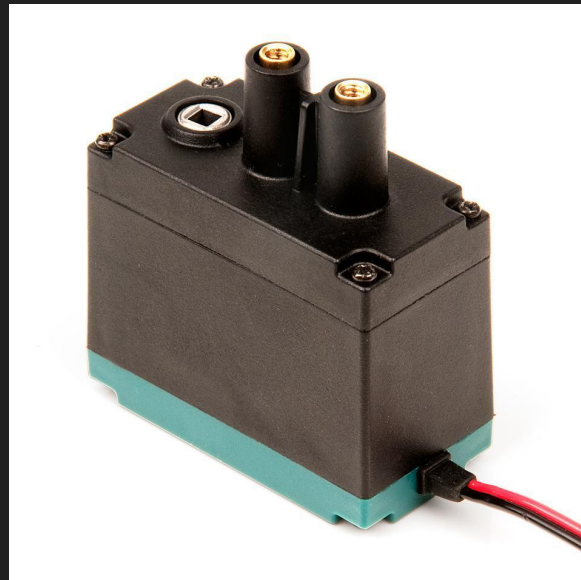


Communicate over VEXnet

In order to use VEXnet, the two devices must be paired using the WIRED method previously done.

This will start off from the end of the last instruction. Please note that the specific program JoystickTest requires the use of one Two-Wire motor as pictured, but motors are not required to communicate between the Cortex and Joystick.

NOTE: Joystick must have batteries for wireless communications!



Communicate over VEXnet

In order to change from Wired to Wireless (VEXnet)
we must now use 2 of the VEXnet USB wifi keys

**NOTE: Make sure the Cortex
and Controller are turned off!**



Communicate over VEXnet

Unplug the USB cable from the Cortex and the Joystick.
Insert the USB Wifi keys in their place (as pictured)



NOTE: This is the old version, you should be using two WHITE usb keys.



Communicate over VEXnet

Turn both the Cortex and Joystick on - after about 5 seconds, the VEXnet light should be fastly blinking green if it works, otherwise try again. If it works, you are done!

Upload Code over Joystick

Alternatively, you can program the Cortex by plugging the Joystick into the computer and downloading code through it. This is useful if you have your cortex on a robot and it is difficult to reach the USB port. Please note, the Joystick must be connected to the Cortex either by USB cable, or a working VEXnet link.

Upload Code over Joystick

On top of a fully functional Cortex and Joystick that communicate with each other, you will also need the Programming Cable set:



Upload Code over Joystick

Plug the RJ12 (see picture A) into the Programming port on the Joystick (see picture B). Plug the USB end into the computer while EasyC is running. Make sure both the Joystick and Cortex are powered, and then download code as normal.

A

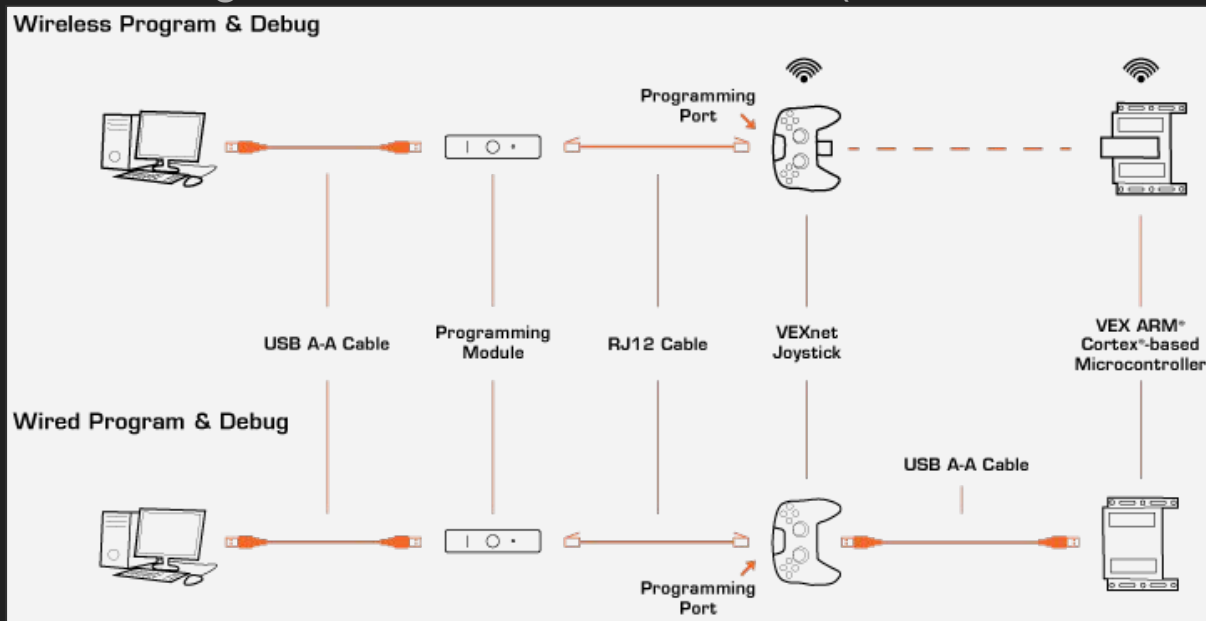


B



Upload Code over Joystick

NOTE: While connected to the Joystick, you can still receive messages in the Terminal from the Cortex as long as the Cortex is connected (over VEXnet or USB) to the Joystick.



fin

Index

- Title Page
- Learning Objectives
- Terms & Abbreviations
- USB Types
- Navigating the IDE
- Recommended Settings
- Download Firmware to Robot
- Download Firmware to Joystick
- Test Cortex
- Upload Sample Code
- Communicate over wire
- Communicate over VEXnet
- Upload code over Joystick